A Trust Network Model Based on Hesitant Fuzzy Linguistic Term Sets

Jieyu Zhan[1], Yuncheng Jiang[1⋆], Wenjun Ma[1], Xudong Luo[2], and Weiru Liu[3]

[1] School of Computer Science, South China Normal University, China
[2] Department of Information and Management Science, Guangxi Normal University, China
[3] School of Computer Science, Electrical and Electronic Engineering, and Engineering Maths, University of Bristol, UK

Abstract. Trust evaluation in a network is important in many areas, such as group decision-making and recommendation in e-commerce. Hence, researchers have proposed various trust network models, in which each agent rates the trustworthiness of others. Most of the existing work require the agents to provide accurate degrees of trust and distrust in advance. However, humans usually hesitate to choose one among several values to assess the trust in another person and tend to express the trust through linguistic descriptions. Hence, this paper proposes a novel trust network model that takes linguistic expression of trust into consideration. More specifically, we structure trust scores based on hesitant fuzzy linguistic term sets and give a comparison method. Moreover, we propose a trust propagation method based on the concept of computing with words to deal with trust relationships between indirectly connected agents, and such a method satisfies some intuitive properties of trust propagation. Finally, we confirm the advantages of our model by comparing it with related work.

Keywords: Trust network · Trust propagation · Hesitant fuzzy linguistic term sets · Concatenation operator · Aggregation operator

1 Introduction

Trust between people is one of the most important factors that influence people's decision making. For example, in the group decision-making problem, the trust relationships between experts can be considered as a reliable source about the importance of the experts [3, 6, 13, 14]. Also, in e-commerce, consumers often know very little about providers of goods or services, but they can choose providers through the recommendations of the people who they trust [7]. Trust relationships in offline communities are often based on face-to-face social experiences. However, the evaluation of trust between users in online social communities should be done through an efficient computational model because of the

⋆ Corresponding author: ycjiang@scnu.edu.cn

lack of direct communication between users [9]. Therefore, various trust network models in which each agent rates the trustworthiness of others have been proposed to describe the trust relationships between users [1, 4, 11].

One of the main issues in the study of trust networks is how to assess the degree to which one agent trusts in another [10]. In most studies, trust is often expressed by accurate numerical values. Nevertheless, these representations cannot reflect well the uncertainty of trust. On the one hand, in real life, because of the different ways of obtaining information, a person may hesitate to choose one among several values to assess the trust in another person. On the other hand, people are more likely to use linguistic descriptions to express their trust. Hence, the uncertainty of assessing trust is also expressed as the ambiguity of language [8]. However, the current representations of trust are often quantitative, which are not always the case in real life. Since online social networks are usually larger than offline social circles, it is difficult for an agent to establish direct trust with some agents. More often, trust relationships between two agents are indirect. Hence, how to infer the trust relationship among indirectly connected agents is an important problem that needs to be solved. However, most of the current trust propagation methods are limited to the quantitative trust score expression, rather than vague linguistic descriptions ones [2, 10, 14].

To tackle the above problems, this paper firstly will propose a novel trust model that takes linguistic expression of trust into consideration. Then, after introducing the concept of hesitant fuzzy linguistic term set (HFLTS) based trust model, we will propose a new method to compare different trust scores based on HFLTS. Further, we will propose a trust propagation mechanism to evaluate trust between two indirectly connected agents.

The main contributions of this paper can be summarized as follows. (1) We propose a trust model based on hesitant fuzzy linguistic term set to better reflect the uncertainty and ambiguity of trust. (2) For indirectly connected agents in trust networks, we propose a trust propagation method based on computing with word methodology to deal with new expression of trust.

The rest of this paper is organised as follows. Firstly we recap the basic concepts and notations about hesitant fuzzy linguistic term sets. Then we propose trust score based on HFLTS and their comparison. Next we present two operators in trust propagation mechanism and reveal some properties of the operators. Finally we discuss the related work and draw our conclusion of the paper.

2 Preliminaries

Human usually use linguistic terms for modeling performance evaluations, such as the word *low*, *medium*, *high* and so on. The linguistic term set (LTS) can be defined as an ordered structure providing the term set that is distributed on a scale on which a total order is defined [15]. Taking a linguistic term set with seven terms, S, as an example, it could be given as follows:

$$S=\{s_0(\text{nothing}), s_1(\text{very low}), s_2(\text{low}), s_3(\text{medium}), s_4(\text{high}), s_5(\text{very high}), s_6(\text{perfect})\}.$$

And the following additional characteristics should be satisfied:

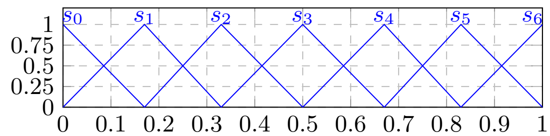

Fig. 1. Membership functions of linguistic terms

(a) negation operator: $\text{neg}(s_i) = s_j$ so that $j = g - i$ ($g + 1$ is the granularity of the term set);
(b) maximisation operator: $\max(s_i, s_j) = s_i$ if $s_i \geq s_j$; and
(c) minimisation operator: $\min(s_i, s_j) = s_i$ if $s_i \leq s_j$.

In this paper, we assign each linguistic term s_i with a triangular membership function as its semantics, which can be represented as $s_i = (a, b, c)$ (a, b and c are parameters of the membership function), because this type of membership function can not only express the ambiguity of linguistic terms, but also reduce the complexity of later calculation about computing with word. More specifically, their semantics can be graphically represented by Fig. 1, where $s_0 = (0, 0, 0.17)$, $s_1 = (0, 0.17, 0.33)$, $s_2 = (0.17, 0.33, 0.5)$, $s_3 = (0.33, 0.5, 0.67)$, $s_4 = (0.5, 0.67, 0.83)$, $s_5 = (0.67, 0.83, 1)$, and $s_6 = (0.83, 1, 1)$.

A linguistic variable represents a variable whose values are words or sentences. For example, *age* is a linguistic variable if its values are linguistic, such as *young*, *not young* and so on [16], and we can also regard *trust* as a linguistic variable. However, in some situation, people may hesitate to choose one among several values to assign to linguistic variable. To this end, the concept of *hesitant fuzzy linguistic term sets* (HFLTSs) is introduced as follows [8]:

Definition 1. *Let S be a linguistic term set, $S = \{s_0, \ldots, s_g\}$, an HFLTS, denoted as h_S, is an ordered finite subset of the consecutive linguistic terms of S. The set of all HFLTSs based on the S is denoted by H_S.*

Example 1. Let $S = \{s_0(\text{nothing}), s_1(\text{very low}), s_2(\text{low}), s_3(\text{medium}), s_4(\text{high}), s_5(\text{very high}), s_6(\text{perfect})\}$ be a linguistic term set, then $h_S^1 = \{s_1(\text{very low}), s_2(\text{low}), s_3(\text{medium})\}$ and $h_S^2 = \{s_3(\text{medium}), s_4(\text{high}), s_5(\text{very high})\}$ are both HFLTSs.

As the expression of HFLTS is still not natural enough, a context-free grammar approach is proposed to generate linguistic term sets [8].

Definition 2. *Let $G_H = (V_N, V_T, I, P)$ be a context-free grammar, and $S = \{s_0, \ldots, s_g\}$ be a linguistic term set. The elements of G_H are defined as follows:*

– V_N = {*<primary term>, <composite term>, <unary relation>, <binary relation>, <conjunction>*};
– V_T = {*lower than, greater than, at least, at most, between, and,* $s_0, s_1, ..., s_g$};
– $I \in V_N$;
– P = {*I ::=<primary term> | <composite term>*
<composite term>::=<unary relation> <primary term> | <binary relation> <primary term><conjunction><primary term>
<primary term>::= $s_0|s_1|...|s_g$
<unary relation>::=lower than|greater than|at least|at most
<binary relation>::= between
<conjunction>::= and},

where the brackets enclose optional elements and the symbol | indicates alternative elements.

Example 2. Let S be the same as that in Example 1. The following linguistic information (denoted as ll) can be obtained by the context-free grammar G_H, such as $ll_1 =$ very low, $ll_2 =$ lower than medium, $ll_3 =$ between low and high.

The following definition [8] transforms the linguistic expressions produced by G_H into HFLTSs.

Definition 3. *Let S_{ll} be the set of linguistic expressions ll produced by G_H, H_S be the set of HFLTSs, $E_{G_H} : S_{ll} \to H_S$ be a function that transforms linguistic expressions into HFLTSs, the linguistic expressions $ll \in S_{ll}$ are transformed into HFLTSs in different ways according to their meanings:*
- $E_{G_H}(s_i) = \{s_i \mid s_i \in S\}$;
- $E_{G_H}(\textit{less than } s_i) = \{s_j \mid s_j \in S \textit{ and } s_j < s_i\}$;
- $E_{G_H}(\textit{greater than } s_i) = \{s_j \mid s_j \in S \textit{ and } s_j > s_i\}$;
- $E_{G_H}(\textit{at least } s_i) = \{s_j \mid s_j \in S \textit{ and } s_j \geq s_i\}$;
- $E_{G_H}(\textit{at most } s_i) = \{s_j \mid s_j \in S \textit{ and } s_j \leq s_i\}$;
- $E_{G_H}(\textit{between } s_i \textit{ and } s_j) = \{s_k \mid s_k \in S \textit{ and } s_i \leq s_k \leq s_j\}$.

Next, we recall the concept of uninorm operator.

Definition 4. *A binary operator* $\uplus : [0,1] \times [0,1] \to [0,1]$ *is a uninorm operator that satisfies increasing, associative and commutative properties, and there exists* $\tau \in [0,1]$, *s.t.*

$$\forall a \in [0,1], a \uplus \tau = a, \tag{1}$$

where τ is said to be the unit element of a uninorm.

3 Trust and Distrust Modeling in Trust Networks

In this section, we will discuss how to model the trust and distrust degree in trust network, and how to compare different trust scores based on HFLTS.

3.1 Trust and Distrust Assessment

Definition 5. *A trust network is a 3-tuple (A,E,R), where*
- *A is the set of agents in the network;*
- *E is the set of trust connections (x, y), where agents $x, y \in A$; and*
- *R is a function $E \to T$, where $R(x, y)$ is called the trust score of agent x in agent y and T is the set of trust scores.*

A trust network assigns scores for pairs of agents reflecting the opinions of agents about each other. In order to be close to the use of natural language, we use linguistic expressions to represent trust scores.

Definition 6. *Given S_t as a linguistic term set of trust and S_d as a linguistic term set of distrust, a trust score defined in Definition 5 is a pair (t, d), where*
- *$t \in S_{ll_t}$ is a trust degree, where S_{ll_t} is the set of linguistic expressions of trust produced by G_H with S_t.*

– *$d \in S_{ll_d}$ is a distrust degree, where S_{ll_d} is the set of linguistic expressions of distrust produced by G_H with S_d.*

Example 3. Let x and y be two directly connected agents in the trust network,

$$\begin{aligned} S_t = \{&s_0(\text{no trust}), s_1(\text{very low trust}), s_2(\text{low trust}), s_3(\text{medium trust}),\\ &s_4(\text{high trust}), s_5(\text{very high trust}), s_6(\text{complete trust})\}\text{and}\\ S_d = \{&s'_0(\text{no distrust}), s'_1(\text{very low distrust}), s'_2(\text{low distrust}), s'_3(\text{medium distrust}),\\ &s'_4(\text{high distrust}), s'_5(\text{very high distrust}), s'_6(\text{complete distrust})\} \end{aligned}$$

be a linguistic term set of trust and a linguistic term set of distrust, rspectively, then the trust relationship between x and y can be represented as follows:

– $R(x, y) =$ (greater than medium trust, low distrust), and
– $R(y, x) =$ (low trust, at least high distrust).

According to Definition 3, the trust scores in Example 3 can be transformed into the following expressions:

– $R(x, y) =$ ({high trust,very high trust, complete trust}, {low distrust}), and
– $R(y, x) =$ ({low trust}, {high distrust,very high distrust, complete distrust}).

3.2 Comparison of Trust Score

For an agent in the network, how to select a more trusted agent in the same network is one of the most important issues. Hence we present a method for comparison. Since the two components of trust score, trust degree and distrust degree, can be transforms into HFLTS in our model, we use h_{S_t} and h_{S_d} to represent the trust degree and distrust degree, respectively. Such two dimensions should be considered when comparing trust scores. The intuition is that the higher the trust degree and the lower the distrust degree, the higher trust scores for another agent. Since both trust and distrust degrees are represented by HFLTSs, we first need a method of comparing HFLTSs. Formally, we have:

Definition 7. *Given two HFLTSs h^1_S and h^2_S, the dominance degree of h^1_S to h^2_S is defined as follows:*

$$D(h^1_S, h^2_S) = \sum_{s^1_i \in h^1_S} \|\{s^2_i \in h^2_S\} \mid s^1_i \geq s^2_i\|, \tag{2}$$

where $\|S\|$ is the cardinality of set S.

The above definition means when the elements in the both HFLTSs are compared in pairs, the number of undominanted pairs is used as the dominance degree of one set to the other.

We present a method for comparison as follows:

Definition 8. *Given $R(x, y) = (h^1_{S_t}, h^1_{S_d})$, and $R(x, z) = (h^2_{S_t}, h^2_{S_d})$, $R(x, y) \geq R(x, z)$ if $D(h^1_{S_t}, h^2_{S_t}) \geq D(h^2_{S_t}, h^1_{S_t})$ and $D(h^2_{S_d}, h^1_{S_d}) \geq D(h^1_{S_d}, h^2_{S_d})$.*

The abvoe definition means that for two trust scores, if one's trust degree is not lower than the other's and the distrust degree is not higher than the other's, then this trust score is not lower than the other trust score. Next we reveal some properties of this method. Firstly, we introduce the concepts of upper bound and lower bound of HFLTS.

Definition 9. *The upper bound h_{S^+} and lower bound h_{S^-} of the HFLTS h_S are defined as follows:*

- $h_{S^+} = \max(s_i) = s_j, \forall s_i \in h_S, s_i \leq s_j$;
- $h_{S^-} = \min(s_i) = s_j, \forall s_i \in h_S, s_i \geq s_j$.

The following theorem states that when all the linguistic terms of trust degree in a trust score are not lower than any linguistic terms of the other's and all the linguistic terms of distrust degree are not higher than any linguistic terms of the other's, then the trust score is definitely not lower than the other one.

Theorem 1 (Complete Domination). *Given $R(x,y) = (h^1_{S_t}, h^1_{S_d})$, and $R(x,z) = (h^2_{S_t}, h^2_{S_d})$, if $h^1_{S_t^-} \geq h^2_{S_t^+}$ and $h^1_{S_d^+} \leq h^2_{S_d^-}$, then $R(x,y) \geq R(x,z)$.*

Proof. Because $h^1_{S_t^-} \geq h^2_{S_t^+}$ and $h^1_{S_d^+} \leq h^2_{S_d^-}$, then $\forall (s^1_i, s^1_j) \in h^1_{S_t} \times h^1_{S_d}, \forall (s^2_i, s^2_j) \in h^2_{S_t} \times h^2_{S_d}, s^1_i \geq s^2_i$ and $s^1_j \leq s^2_j$. By formula (2), we have:

$$D(h^1_{S_t}, h^2_{S_t}) = \sum_{s^1_i \in h^1_{S_t}} \|\{s^2_i \in h^2_{S_t}\} \mid s^1_i \geq s^2_i\| = \|h^1_{S_t}\| \cdot \|h^2_{S_t}\|,$$

$$D(h^2_{S_t}, h^1_{S_t}) = \begin{cases} 1 & \text{if } h^1_{S_t^-} = h^2_{S_t^+}, \\ 0 & \text{othewise.} \end{cases}$$

Hence, $D(h^1_{S_t}, h^2_{S_t}) \geq D(h^2_{S_t}, h^1_{S_t})$. Similarly, we have

$$D(h^2_{S_d}, h^1_{S_d}) = \|h^1_{S_d}\| \cdot \|h^2_{S_d}\|,$$

$$D(h^1_{S_d}, h^2_{S_d}) = \begin{cases} 1 & \text{if } h^1_{S_d^+} = h^2_{S_d^-}, \\ 0 & \text{othewise.} \end{cases}$$

Hence, $D(h^2_{S_d}, h^1_{S_d}) \geq D(h^1_{S_d}, h^2_{S_d})$. According to Definiton 8, we have $R(x,y) \geq R(x,z)$. □

The following theorem reveals the monotonicity of trust score.

Theorem 2 (Monotonicity). *Given $S_t = \{s^1_0, \ldots, s^1_{N-1}\}, S_d = \{s^2_0, \ldots, s^2_{M-1}\}$, $R(x,y) = (h_{S_t}, h_{S_d})$, where*

$$h_{S_t} = \{s^1_k, s^1_{k+1}, \ldots, s^1_{k+n-1}\}, 0 \leq k \leq N-n,$$
$$h_{S_d} = \{s^2_g, s^2_{g+1}, \ldots, s^2_{g+m-1}\}, 0 \leq g \leq M-m,$$

where N, M, n and m are the the cardinality of S_t, S_d, h_{S_t} and h_{S_d}, respectively. Let $R^1(x,y) = (h'_{S_t}, h_{S_d})$, $R^2(x,y) = (h_{S_t}, h'_{S_d})$, $R^3(x,y) = (h''_{S_t}, h_{S_d})$, $R^4(x,y) = (h_{S_t}, h''_{S_d})$, in which

$$h'_{S_t} = \{s^1_k, s^1_{k+1}, \ldots, s^1_{k+n} \mid \ k+n \leq N-1\},$$
$$h'_{S_d} = \{s^2_{g-1}, s^2_g, \ldots, s^2_{g+m-1} \mid g-1 \geq 0\},$$
$$h''_{S_t} = \{s^1_{k-1}, s^1_k, \ldots, s^1_{k+n-1} \mid k-1 \geq 0\},$$
$$h''_{S_d} = \{s^2_g, s^2_{g+1}, \ldots, s^2_{g+m} \mid g+m \leq M-1\},$$

then (1) $R^1(x,y) \geq R(x,y)$, (2) $R^2(x,y) \geq R(x,y)$, (3) $R(x,y) \geq R^3(x,y)$, and (4) $R(x,y) \geq R^4(x,y)$.

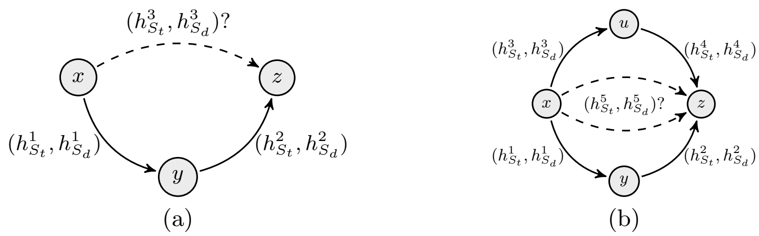

Fig. 2. Trust propagation in a trust network

Proof. (1) Because $\forall s_i^2 \in h_{S_t}, s^1_{k+n} > s_i^2$, we have

$$\begin{aligned}&D(h'_{S_t}, h_{S_t})\\ &= D(h_{S_t}, h_{S_t}) + \|\{s_j^2 \in h_{S_d}\} \mid s^1_{k+n} \geq s_j^2\|\\ &= D(h_{S_t}, h_{S_t}) + \|h_{S_t}\|,\end{aligned}$$

and

$$D(h_{S_t}, h'_{S_t}) = D(h_{S_t}, h_{S_t}).$$

Hence, $D(h'_{S_t}, h_{S_t}) \geq D(h_{S_t}, h'_{S_t})$. Because $D(h_{S_d}, h_{S_d}) \geq D(h_{S_d}, h_{S_d})$, by Definition 8, we have $R^1(x, y) \geq R(x, y)$. Similarly, the other situations (2), (3) and (4) can be proved. □

4 Trust Propagation Method

In this section, we propose a trust propagation method that can be used in HFLTS based trust model. It consists of two important components: one is used to propagate the trust score along a path in the trust network that connects two indirectly connected agents with trusted third agents (see Fig. 2(a)); and the other one is used to aggregate the assessments in different trust paths, because there may be multiple paths to access the same indirectly connected agent (see Fig. 2(b)). Hence, we define two operators: concatenation operator $\otimes$ and aggregation operator $\oplus$, to solve the above problems, respectively. Finally, we evaluate the trust propagation method and reveal some properties of the operators.

4.1 Concatenation Operator

Definition 10. *Given two triangular membership functions of linguistic terms $s_1 = (a_1, b_1, c_1)$ and $s_2 = (a_2, b_2, c_2)$, the aggregation of s_1 and s_2, denoted as s_3 in a linguistic term set S, is defined as $s_3 = app_S({s_3}')$, where*

$$ {s_3}' = s_1 \diamond s_2 = (a_1 \cdot a_2, b_1 \cdot b_2, c_1 \cdot c_2), \tag{3}$$

and $app_S(s_i)$ is a linguistic approximation process, which is used to seclect a linguistic term $s^\star$ in S that has minimum distance to s_i, i.e., $\forall s_j \in S, d(s^\star, s_i) \leq d(s_j, s_i)$, where d() is the distance between two membership functions.

We adopt Euclidean distance in this paper, which is defined as follows:

Definition 11. *Give* $s_i = (a_i, b_i, c_i)$ *and* $s_j = (a_j, b_j, c_j)$,

$$d(s_j, s_i) = \sqrt{p_1(a_i - a_j)^2 + p_2(b_i - b_j)^2 + p_3(c_i - c_j)^2}, \quad (4)$$

where p_i $(i \in \{1, 2, 3\})$ *measures the representativeness of parameters of membership functions. We set* $p_1 = 0.1$, $p_2 = 0.8$ *and* $p_3 = 0.1$, *because for a triangular membership function* $s = (a, b, c)$, b *is the most representative component.*

Definition 12 (Trust Concatenation Operator). *Suppose in a trust network* (A, E, R), $(x, y), (y, z) \in E$, *while* $(x, z) \notin E$, $R(x, y) = (h^1_{S_t}, h^1_{S_d})$ *and* $R(y, z) = (h^2_{S_t}, h^2_{S_d})$, *then*

$$(h^3_{S_t}, h^3_{S_d}) = (h^1_{S_t}, h^1_{S_d}) \otimes (h^2_{S_t}, h^2_{S_d})$$

where

$$h^3_{S_t^+} = app_{S_t}(h^1_{S_t^+} \diamond h^2_{S_t^+}), h^3_{S_t^-} = app_{S_t}(h^1_{S_t^-} \diamond h^2_{S_t^-}),$$
$$h^3_{S_d^+} = app_{S_d}(h^1_{S_t^+} \diamond h^2_{S_d^+}), h^3_{S_d^-} = app_{S_d}(h^1_{S_t^-} \diamond h^2_{S_d^-}),$$

with $\otimes$ *is a concatenation operator.*

Example 4. Given the semantics of linguistic term sets of trust and distrust, S_t and S_d as in Example 3 with membership functions as in Fig. 1, and a trust network as shown in Fig. 2(a), in which agents x and y are connected, y and z are connected, while x and z are indirectly connected, $R(x, y) = (h^1_{S_t}, h^1_{S_d})$, $R(y, z) = (h^2_{S_t}, h^2_{S_d})$, where $h^1_{S_t} = \{s_4, s_5\}$, $h^1_{S_d} = \{s'_0, s'_1\}$, $h^2_{S_t} = \{s_5\}$, and $h^2_{S_d} = \{s'_2, s'_3\}$, we can evaluate the trust of x in z based on concatenation operator $\otimes$.

Now we see how to calculate the trust score $R(x, y) = (h^3_{S_t}, h^3_{S_d})$. Specifically, we should calculate the upper bounds and lower bounds of $h^3_{S_t}$ and $h^3_{S_d}$, respectively. We take $h^3_{S_t^+}$ as an example. By formula (3), we have

$$h^{3'}_{S_t^+} = h^1_{S_t^+} \diamond h^2_{S_t^+} = s_5 \diamond s_5 = (0.67 \cdot 0.67, 0.83 \cdot 0.83, 1 \cdot 1) = (0.4489, 0.6889, 1).$$

By formula (4), we have

$$d(s_6, h^{3'}_{S_t^+}) = \sqrt{0.1(0.83 - 0.4489)^2 + 0.8(1 - 0.6889)^2 + 0.1(1 - 1)^2} = 0.3032.$$

Similarly, we can obtain

$$d(s_5, h^{3'}_{S_t^+}) = 0.1443, d(s_4, h^{3'}_{S_t^+}) = 0.0586, d(s_3, h^{3'}_{S_t^+}) = 0.2021,$$
$$d(s_2, h^{3'}_{S_t^+}) = 0.3685, d(s_1, h^{3'}_{S_t^+}) = 0.5296, d(s_0, h^{3'}_{S_t^+}) = 0.6846.$$

Hence, $h^3_{S_t^+} = app_{S_t}(h^{3'}_{S_t^+}) = s_4$. Similarly, we can obtain $h^3_{S_t^-} = app_{S_t}(s_4 \diamond s_5) = s_3$, $h^3_{S_d^+} = app_{S_d}(s_5 \diamond s'_3) = s'_3$, and $h^3_{S_d^-} = app_{S_d}(s_4 \diamond s'_2) = s'_1$, then $R(x, z) = (\{s_3, s_4\}, \{s'_1, s'_2, s'_3\})$.

From the results we can see that compared with the trust degrees and distrust degrees of x in y and y in z, those of x in z decrease slightly. This is similar to our intuition that as agents' distance increases, the degree of trust will decrease.

4.2 Aggregation Operator

There may be more than one path to propagate trust scores from an agent to an indirectly connected agent in a trust network. Therefore, we need a method that aggregates the trust scores passed through different trust paths to have an overall understanding of the unknown agent.

When aggregating different trust scores, it is necessary to conform to such intuitions: when an agent obtains high degrees of trust through different trust paths, its trust in the unknown agent should be strengthened; when the results obtained are low degrees of trust, its trust in the unknown agent should be weakened; and when the results obtained are in conflict, the aggregated one is a compromise. Hence, when aggregating several triangular membership functions of linguistic terms in the aggregation phase, we employ a uninorm operator to aggregate the parameters of membership functions. Formally, we have:

Definition 13. *Given n triangular membership functions of linguistic terms $s_i = (a_i, b_i, c_i), i \in \{1, \dots, n\}$, the aggregation of these linguistic terms, denoted as s in a linguistic term set S, is defined as $s = app_S(s^{'})$, where*

$$s^{'} = \blacktriangle_{i=1}^{n} s^i = (a_1 \uplus_1 a_2 \uplus_1 \cdots \uplus_1 a_n, b_1 \uplus_2 b_2 \uplus_2 \cdots \uplus_2 b_n, c_1 \uplus_3 c_2 \uplus_3 \cdots \uplus_3 c_n), \tag{5}$$

where $\uplus_i$ is give by:

$$x \uplus_i y = \begin{cases} 0.5 & \text{if } (x,y) = (1,0) \text{ or } (x,y) = (0,1), \\ \frac{(1-\tau_i)xy}{(1-\tau_i)xy + \tau_i(1-x)(1-y)} & \text{otherwise,} \end{cases}$$

and $app_S(s_i)$ is the same as that in Definition 10.

Definition 14 (Trust Aggregation Operator). *Suppose in a trust network (A, E, R), $(x, y) \notin E$, while there are n paths to propagate trust from x to y, and the propagation results are $R^{path_i}(x, y) = (h_{S_t}^{path_i}, h_{S_d}^{path_i}), i \in \{1, \dots, n\}$, the aggregation of trust scores along all paths, denoted as $R(x, y) = (h_{S_t}, h_{S_d})$, is given by:*

$$(h_{S_t}, h_{S_d}) = \otimes_{i=1}^{n} (h_{S_t}^{path_i}, h_{S_d}^{path_i})$$

where

$$h_{S_t^+} = app_{S_t}(\blacktriangle_{i=1}^{n} h_{S_t^+}^{path_i}), h_{S_t^-} = app_{S_t}(\blacktriangle_{i=1}^{n} h_{S_t^-}^{path_i}),$$
$$h_{S_d^+} = app_{S_d}(\blacktriangle_{i=1}^{n} h_{S_d^+}^{path_i}), h_{S_d^-} = app_{S_d}(\blacktriangle_{i=1}^{n} h_{S_d^-}^{path_i})$$

with $\oplus$ is a aggregation operator.

Example 5. Given the semantics of linguistic term sets of trust and distrust, S_t and S_d as in Example 3 with membership functions as in Fig. 1, and a trust network (A, E, R) as shown in Fig. 2(b), in which $(x, y), (y, z), (x, u), (u, z) \in E$, while $(x, z) \notin E$, and $\tau_1 = 0.33, \tau_2 = 0.5, \tau_3 = 0.67$, let $R(x, y) = (h_{S_t}^1, h_{S_d}^1)$, $R(y, z) = (h_{S_t}^2, h_{S_d}^2)$,$R(x, u) = (h_{S_t}^3, h_{S_d}^3), R(u, z) = (h_{S_t}^4, h_{S_d}^4)$, where $h_{S_t}^1 = \{s_4, s_5\}$, $h_{S_d}^1 = \{s'_0, s'_1\}$, $h_{S_t}^2 = \{s_5\}$, $h_{S_d}^2 = \{s'_2, s'_3\}$, $h_{S_t}^3 = \{s_6\}$, $h_{S_d}^3 = \{s'_0, s'_1\}$, $h_{S_t}^4 = \{s_5, s_6\}$, and $h_{S_d}^4 = \{s'_1\}$. Then we can evaluate the trust score of x in z based on both concatenation operator $\otimes$ and aggregation operator $\oplus$.

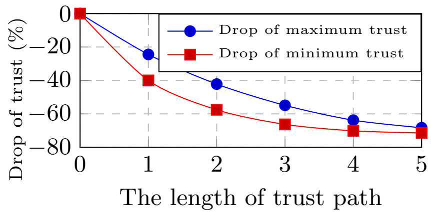

Fig. 3. Drop of trust

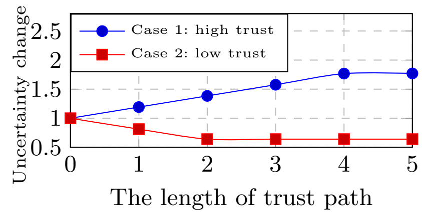

Fig. 4. Uncertainty change of trust degree

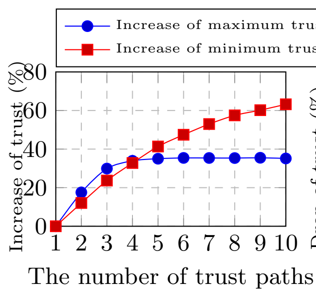

Fig. 5. Increase of trust

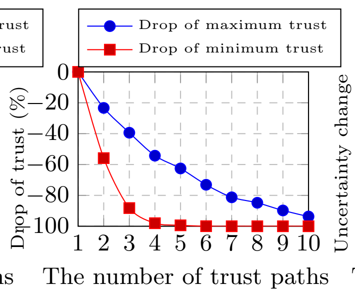

Fig. 6. Drop of trust

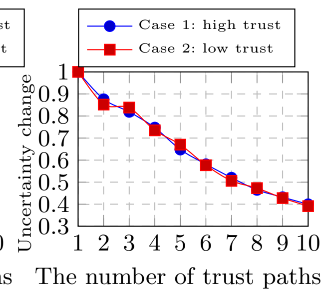

Fig. 7. Uncertainty change

Now we see how to calculate $R(x,z) = (h^5_{S_t}, h^5_{S_d})$. Firstly, we find that there are two trust propagation paths from x to y, that is, "$path_1 : x \to y \to z$" and "$path_2 : x \to u \to z$". Hence, we should first calculate the trust scores along different trust propagation paths separately. According to Definition 12, we have

$$R^{path_1}(x,z) = (h^{path_1}_{S_t}, h^{path_1}_{S_d}) = (\{s_3, s_4\}, \{s'_1, s'_2, s'_3\}),$$
$$R^{path_2}(x,z) = (h^{path_2}_{S_t}, h^{path_2}_{S_d}) = (\{s_5, s_6\}, \{s'_1\}).$$

Secondly, we aggregate the above trust scores along different paths into $R(x,z)$ by Definition 14. We take $h^3_{S^+_t}$ as an example. By formula (5), we have

$$h^{5'}_{S^+_t} = h^{path_1}_{S^+_t} \blacktriangle h^{path_2}_{S^+_t} = s_4 \blacktriangle s_6 = (0.5 \uplus_1 0.83, 0.67 \uplus_2 1, 0.83 \uplus_3 1) = (0.9084, 1, 1).$$

By formula (4), we obtain $d(s_6, h^{5'}_{S^+_t}) = 0.0248$, $d(s_5, h^{5'}_{S^+_t}) = 0.1697$, $d(s_4, h^{5'}_{S^+_t}) = 0.3266$, $d(s_3, h^{5'}_{S^+_t}) = 0.4943$, $d(s_2, h^{5'}_{S^+_t}) = 0.6623$, $d(s_1, h^{5'}_{S^+_t}) = 0.8237$, $d(s_0, h^{5'}_{S^+_t}) = 0.9754$. Hence, $h^5_{S^+_t} = app_{S_t}(h^{5'}_{S^+_t}) = s_6$. Similarly, we can obtain

$$h^5_{S^-_t} = app_{S_t}(s_3 \blacktriangle s_5) = s_5, h^5_{S^+_d} = app_{S_d}(s'_3 \blacktriangle s'_1) = s'_1, h^5_{S^-_d} = app_{S_d}(s'_1 \blacktriangle s'_1) = s'_0.$$

Hence, $R(x,z) = (\{s_5, s_6\}, \{s'_0, s'_1\})$. This result is in line with intuition that when the information obtained through two paths are both relatively high trust and low distrust, the aggregated result should strengthen the original degrees.

4.3 Evaluation

This section conducts two experiments to reveal some insight into our method.

The first experiment is conducted to see how the operator influences the trust degree and uncertainty of trust degree when the length of trust path changes. We randomly generate an HFLTS based trust degree from the semantics of linguistic term sets of trust S_t as in Example 3 with membership functions as in Fig. 1. We suppose the agents have the same trust degree along a trust path with n agents ($n-1$ is the length of trust path). We use the ratio of original trust degree and final one to represent changes of trust and use the number of elements in an HFLTS based trust degree to represent the uncertainty of trust degree. We run the calculation of propagation results 1,000 times under the above setting. From Fig. 3, we can that see no matter what the original trust degree is, both the maximum and minimum trust degrees drop when the length of trust path increases. However, there are two different trends of uncertainty changes. Fig. 4 shows that if the trust in propagation is high, then the uncertainty will increase, because the trust degree is weakening in the process of propagation. However, if the trust in propagation is low, the uncertainty of trust degree decreases in the process of propagation and the trust degree will reach a low level quickly.

We carry out the second experiment to see how the aggregation operator influences the trust degree and uncertainty of trust degree as the number of trust paths changes. We randomly generate n (in-between 1 and 10) HFLTS based trust degrees from the semantics of linguistic term sets of trust $\{s_3, s_4, s_5\}$ and $\{s_1, s_2, s_3\}$, respectively. We aggregate the n trust degrees along different paths and run the calculation 1,000 times under the above setting. Fig. 5 shows that when the trust degree of each trust path is relatively high, the trust degree of aggregation will be enhanced, especially when the aggregation route increases, the enhancement effect will be more obvious. However, Fig. 6 shows that when trust degree of each trust path is relatively low, the aggregation results will gradually weaken with the increase of the number of paths. Different from the concatenation operator, in Fig. 7 we can see the uncertainty of aggregated results decreases with the increase of the number of trust paths if all the trust degrees on different trust paths are relatively high or all of them are relatively low.

From the above analysis, we can see that the operators play different roles in trust propagation, and each of them satisfies some intuitive properties of propagation. Therefore, the HFLTS based trust model is not only closer to human users in expression, but also can maintain intuitive trust propagation properties.

5 Related Work

Majd and Balakrishnan [7] propose a trust model, which considers reliability, similarity, satisfaction and trust transitivity. In the trust transitivity step, they also employ the concatenation operator and aggregation operator proposed by [12] to identify the trust value of each suggested recommender. Their model focuses on how to choose more reliable recommender based on identified components, while in our model, we pay more attention to characterising the trust model that is more similar to humans' expression and the trust propagation mechanism based on this kind of model.

Wu et al. [14] propose a trust network model for determining the importance value assigned to the experts in group decision making problem. They investigate a uninorm propagation operator to propagate both trust and distrust information and prevents the loss of trust information. Later, Wu et al. [13] propose a visual interaction consensus model for social network group decision making based on this trust propagation method. We also employ uninorm operator in trust propagation process. However, we use it to aggregate the trust scores passed through different trust paths, and the aggregation operator is used in qualitative trust model, in which the trust scores are represented by linguistic expressions. Moreover, their propagation method only considers the shortest trust path and neglects the others, while ours consider all paths when aggregating trust scores.

The methods of fuzzy logic have been used in some trust network models. For example, Kant and Bharadwaj [5] propose a fuzzy computation based trust model, which is employed in recommender systems to deal with the cold start problem. They also use linguistic expressions to represent trust and distrust concepts and propose relevant propagation and aggregation operators. However, in their model, only one linguistic expression can be assigned to trust and distrust degrees, while our model employ hesitant fuzzy linguistic term sets to reflect that a person may hesitate to choose one among several values to assess the trust.

6 Conclusion

This paper studied how to evaluate trust scores in trust network with linguistic expressions. Firstly, we proposed the representation of trust score based on HFLTS to reflect humans' hesitation and developed a novel method of comparison between trust scores. Secondly, we introduced concatenation operator and aggregation operator to deal with the problem of evaluating trust between indirectly connected agents. Finally, we showed that the operators meet the intuitions of trust propagation. In the future, one of the most interesting thing is to employ the HFLTS based trust model to construct a recommendation system in e-commerce that can improve the accuracy of recommendations.

Acknowledgments

The works described in this paper are supported by the National Natural Science Foundation of China under Grant Nos. 61772210, 61272066 and 61806080; Guangdong Province Universities Pearl River Scholar Funded Scheme (2018); the Project of Science and Technology in Guangzhou in China under Grant No. 201807010043; the key project in universities in Guangdong Province of China under Grant No. 2016KZDXM024; the Doctoral Start-up Project of Natural Science Foundation of Guangdong Province in China under Grant No. 2018A030310529; the Project of Department of Education of Guangdong Province in China under Grant No. 2017KQNCX048; China Postdoctoral Science Foundation under Grant No. 2018M643115; and Humanities and Social Sciences Foundation of Ministry of Education of China (No. 18YJC72040002).

References

1. Agreste, S., De Meo, P., Ferrara, E., Piccolo, S., Provetti, A.: Trust networks: Topology, dynamics, and measurements. IEEE Internet Computing **19**(6), 26–35 (2015)
2. Ghavipour, M., Meybodi, M.R.: Trust propagation algorithm based on learning automata for inferring local trust in online social networks. Knowledge-Based Systems **143**, 307–316 (2018)
3. Herrera-Viedma, E., Cabrerizo, F.J., Kacprzyk, J., Pedrycz, W.: A review of soft consensus models in a fuzzy environment. Information Fusion **17**, 4–13 (2014)
4. Jøsang, A., Hayward, R., Pope, S.: Trust network analysis with subjective logic. In: Proceedings of the 29th Australasian Computer Science Conference. pp. 85–94 (2006)
5. Kant, V., Bharadwaj, K.K.: Fuzzy computational models of trust and distrust for enhanced recommendations. International Journal of Intelligent Systems **28**(4), 332–365 (2013)
6. Liu, Y., Liang, C., Chiclana, F., Wu, J.: A trust induced recommendation mechanism for reaching consensus in group decision making. Knowledge-Based Systems **119**, 221–231 (2017)
7. Majd, E., Balakrishnan, V.: A trust model for recommender agent systems. Soft Computing **21**(2), 417–433 (2017)
8. Rodriguez, R.M., Martinez, L., Herrera, F.: Hesitant fuzzy linguistic term sets for decision making. IEEE Transactions on Fuzzy Systems **20**(1), 109–119 (2012)
9. Ruan, Y., Durresi, A.: A survey of trust management systems for online social communities–trust modeling, trust inference and attacks. Knowledge-Based Systems **106**, 150–163 (2016)
10. Verbiest, N., Cornelis, C., Victor, P., Herrera-Viedma, E.: Trust and distrust aggregation enhanced with path length incorporation. Fuzzy Sets and Systems **202**, 61–74 (2012)
11. Wang, Y., Vassileva, J.: Bayesian network-based trust model. In: Proceedings of the IEEE/WIC International Conference on Web Intelligence. pp. 372–378 (2003)
12. Wang, Y., Singh, M.P.: Trust representation and aggregation in a distributed agent system. In: Proceedings of the 21st AAAI Conference on Artificial Intelligence. pp. 1425–1430 (2006)
13. Wu, J., Chiclana, F., Fujita, H., Herrera-Viedma, E.: A visual interaction consensus model for social network group decision making with trust propagation. Knowledge-Based Systems **122**, 39–50 (2017)
14. Wu, J., Xiong, R., Chiclana, F.: Uninorm trust propagation and aggregation methods for group decision making in social network with four tuple information. Knowledge-Based Systems **96**, 29–39 (2016)
15. Yager, R.R.: An approach to ordinal decision making. International Journal of Approximate Reasoning **12**(3-4), 237–261 (1995)
16. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning-I. Information sciences **8**(3), 199–249 (1975)